NOTES

How to find HCF ?

① Finding HCF by subtraction

#1	#2	#3
12, 15	15, 60	100, 1
12, 3	15, 45	99, 1
9, 3	15, 30	98, 1
6, 3	15, 15	97, 1
3, 3	15, 0	⋮
0, 3		1, 1
		0, 1

② Finding HCF by $\boxed{\text{modular division}}$?

#1	#2	#3
12, 15	15, 60	100, 1
12, 3	15, 0	0, 1
0, 3		

After observing two calculation methods above we found $\boxed{\text{modular division}}$ is the better way of finding $\boxed{\text{HCF}}$

How to find LCM ?

For two numbers a and b, the equation below stands true.

$$a \times b = HCF(a, b) \times LCM(a, b)$$

$$LCM(a, b) = \frac{a \times b}{HCF(a, b)}$$

NOTE: In javascript, $10^9 + 7$ could be written as 10**9 + 7

$2^{10} = 1024$

NOTES

Question 2 Find the power of a number.

Example 1 $N = 5,\ k = 3$ Simple approach

$5 \times 5 \times 5 = 125$

Example 2 $N = 2,\ k = 10$ Binary Exponentiation.

$2^{10} = (2^2)^5 = (2 \times 2)^5 = 4^5$

$4^5 = 4 \times 4^4$ $\quad 4 \times 256 = 1024$

$4^4 = \cancel{(4^2)} (4^2)^2 = (4 \times 4)^2 = 16^2$

$16^2 = (16 \times 16)^1 = 256^1$

$256^1 = 256 \times \underbrace{256^0}_{1} = 256$

```
double getRoot_pos_neg (double n, double k) {    Program starts here
    double answer = 1;
    double power = k<0 ? k x -1 : k;

    while (power > 0) {

        if (power % 2 == 0) { //if power is even
            n = n x n;
            power = power /2 ;
        }
        else { //if power is odd
            answer = answer x n;
            power = power - 1;
        }
    }
    if (k < 0) {   // is power is negative
        answer = 1 / answer ;
    }
    return answer;
}
```

NOTES

Question 5 Find the no. of trailing 0s in N! i.e. N factorial.
Leading question of A.th magical no. BS2 Que4

5! = 5×4×3×2×1 = 120, There is one trailing zero.

Answer NOTE: Good problem to understand logrithmic complexity.

Approaches

① **Linearly calculating factorial**

For 20! first we will calculate 20! literally and after that we will count the zeroes in the end.

Eg 20! = 20×19×18×17×.....×3×2×1 = 2439.....6640000 (19 digits)
There are 4 zeroes in the end.

But it is not possible to keep factorial value of big integers.

② **5 and 2 counter method.**

We know that whenever we multiply 5 and 2 it returns 10. So, we can count their occurrences to find the no. of trailing 0s in a number's factorial.

Eg 10! = 10 × 9 × 8 × 7 × 6 × 5 × 4 × 3 × 2 × 1
10 → 2.5, 8 → 2.2.2, 6 → 2.3, 5 → 5, 4 → 2.2, 2 → 2

no. of 2 = 8
no. of 5 = 2
Min (no. of 2s, no. of 5s) = 2

NOTES

Now, we know from the example above that we will always have ample of 2s in any factorial calculation because every second number is even. So, we can calculate only no. of 5s in any factorial and that will be the answer.

$\boxed{10}$

5 x 2

Eg.

$15! = 15 \times 14 \times 13 \times 12 \times 11 \times 10 \times 9 \times 8 \times 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1$

3.5   2.5   5

There are 3 fives, so there will be 3 zeroes in the end.

To simplify is further, we will divide N by 5 to get the answer.

$\left\lfloor \frac{15}{5} \right\rfloor = 3$, 3 trailing zeroes in 15!

But for the big integers like 100!, 200! etc. things will change further.

$\left\lfloor \frac{100}{5} \right\rfloor = 20$, but 100! has more than 20 trailing zeroes.

$100! = 1 \times \dots \times 5 \times \dots \times 25 \times \dots \times 50 \times \dots \times 75 \times \dots \times 100$

5.5   5.5.2   5.5.3   5.5.4

Now, we have 4 more 5s available at 25, 50, 75 and 100. So that means 100! has 24 trailing 0s instead of 20. So, calculation will update like below.

※ $\left\lfloor \frac{100}{5} \right\rfloor + \left\lfloor \frac{100}{5^2} \right\rfloor + \left\lfloor \frac{100}{5^3} \right\rfloor$

$20 + 4 + 0 = 24$

IMPORTANT: We can set elements of an array in N! ways where N is the length of the array. So probablity of getting fully sorted array is 1/N!

NOTES

So, we will keep on performing floor division till the time we will not get the result of any floor division as 0. Simultaneously increasing the power of 5. Like in the last example. #

$TC = O(\log_5(n))$

$SC = O(1)$